positional arguments:

file\_or\_dir

general:

-k EXPRESSION only run tests which match the given substring

expression. An expression is a python evaluatable

expression where all names are substring-matched

against test names and their parent classes. Example:

-k 'test\_method or test\_other' matches all test

functions and classes whose name contains

'test\_method' or 'test\_other', while -k 'not

test\_method' matches those that don't contain

'test\_method' in their names. Additionally keywords

are matched to classes and functions containing extra

names in their 'extra\_keyword\_matches' set, as well as

functions which have names assigned directly to them.

-m MARKEXPR only run tests matching given mark expression.

example: -m 'mark1 and not mark2'.

--markers show markers (builtin, plugin and per-project ones).

-x, --exitfirst exit instantly on first error or failed test.

--maxfail=num exit after first num failures or errors.

--strict marks not registered in configuration file raise

errors.

-c file load configuration from `file` instead of trying to

locate one of the implicit configuration files.

--continue-on-collection-errors

Force test execution even if collection errors occur.

--rootdir=ROOTDIR Define root directory for tests. Can be relative path:

'root\_dir', './root\_dir', 'root\_dir/another\_dir/';

absolute path: '/home/user/root\_dir'; path with

variables: '$HOME/root\_dir'.

--fixtures, --funcargs

show available fixtures, sorted by plugin appearance

(fixtures with leading '\_' are only shown with '-v')

--fixtures-per-test show fixtures per test

--import-mode={prepend,append}

prepend/append to sys.path when importing test

modules, default is to prepend.

--pdb start the interactive Python debugger on errors or

KeyboardInterrupt.

--pdbcls=modulename:classname

start a custom interactive Python debugger on errors.

For example:

--pdbcls=IPython.terminal.debugger:TerminalPdb

--trace Immediately break when running each test.

--capture=method per-test capturing method: one of fd|sys|no.

-s shortcut for --capture=no.

--runxfail run tests even if they are marked xfail

--lf, --last-failed rerun only the tests that failed at the last run (or

all if none failed)

--ff, --failed-first run all tests but run the last failures first. This

may re-order tests and thus lead to repeated fixture

setup/teardown

--nf, --new-first run tests from new files first, then the rest of the

tests sorted by file mtime

--cache-show show cache contents, don't perform collection or tests

--cache-clear remove all cache contents at start of test run.

--lfnf={all,none}, --last-failed-no-failures={all,none}

change the behavior when no test failed in the last

run or no information about the last failures was

found in the cache

--arraydiff Enable comparison of arrays to reference arrays stored

in files

--arraydiff-generate-path=ARRAYDIFF\_GENERATE\_PATH

directory to generate reference files in, relative to

location where py.test is run

--arraydiff-reference-path=ARRAYDIFF\_REFERENCE\_PATH

directory containing reference files, relative to

location where py.test is run

--arraydiff-default-format=ARRAYDIFF\_DEFAULT\_FORMAT

Default format for the reference arrays (can be 'fits'

or 'text' currently)

reporting:

-v, --verbose increase verbosity.

-q, --quiet decrease verbosity.

--verbosity=VERBOSE set verbosity

-r chars show extra test summary info as specified by chars

(f)ailed, (E)error, (s)skipped, (x)failed, (X)passed,

(p)passed, (P)passed with output, (a)all except pP.

Warnings are displayed at all times except when

--disable-warnings is set

--disable-warnings, --disable-pytest-warnings

disable warnings summary

-l, --showlocals show locals in tracebacks (disabled by default).

--tb=style traceback print mode (auto/long/short/line/native/no).

--show-capture={no,stdout,stderr,log,all}

Controls how captured stdout/stderr/log is shown on

failed tests. Default is 'all'.

--full-trace don't cut any tracebacks (default is to cut).

--color=color color terminal output (yes/no/auto).

--durations=N show N slowest setup/test durations (N=0 for all).

--pastebin=mode send failed|all info to bpaste.net pastebin service.

--junit-xml=path create junit-xml style report file at given path.

--junit-prefix=str prepend prefix to classnames in junit-xml output

--result-log=path DEPRECATED path for machine-readable result log.

collection:

--collect-only only collect tests, don't execute them.

--pyargs try to interpret all arguments as python packages.

--ignore=path ignore path during collection (multi-allowed).

--deselect=nodeid\_prefix

deselect item during collection (multi-allowed).

--confcutdir=dir only load conftest.py's relative to specified dir.

--noconftest Don't load any conftest.py files.

--keep-duplicates Keep duplicate tests.

--collect-in-virtualenv

Don't ignore tests in a local virtualenv directory

--doctest-modules run doctests in all .py modules

--doctest-report={none,cdiff,ndiff,udiff,only\_first\_failure}

choose another output format for diffs on doctest

failure

--doctest-glob=pat doctests file matching pattern, default: test\*.txt

--doctest-ignore-import-errors

ignore doctest ImportErrors

--doctest-continue-on-failure

for a given doctest, continue to run after the first

failure

test session debugging and configuration:

--basetemp=dir base temporary directory for this test run.

--version display pytest lib version and import information.

-h, --help show help message and configuration info

-p name early-load given plugin (multi-allowed). To avoid

loading of plugins, use the `no:` prefix, e.g.

`no:doctest`.

--trace-config trace considerations of conftest.py files.

--debug store internal tracing debug information in

'pytestdebug.log'.

-o OVERRIDE\_INI, --override-ini=OVERRIDE\_INI

override ini option with "option=value" style, e.g.

`-o xfail\_strict=True -o cache\_dir=cache`.

--assert=MODE Control assertion debugging tools. 'plain' performs no

assertion debugging. 'rewrite' (the default) rewrites

assert statements in test modules on import to provide

assert expression information.

--setup-only only setup fixtures, do not execute tests.

--setup-show show setup of fixtures while executing tests.

--setup-plan show what fixtures and tests would be executed but

don't execute anything.

pytest-warnings:

-W PYTHONWARNINGS, --pythonwarnings=PYTHONWARNINGS

set which warnings to report, see -W option of python

itself.

logging:

--no-print-logs disable printing caught logs on failed tests.

--log-level=LOG\_LEVEL

logging level used by the logging module

--log-format=LOG\_FORMAT

log format as used by the logging module.

--log-date-format=LOG\_DATE\_FORMAT

log date format as used by the logging module.

--log-cli-level=LOG\_CLI\_LEVEL

cli logging level.

--log-cli-format=LOG\_CLI\_FORMAT

log format as used by the logging module.

--log-cli-date-format=LOG\_CLI\_DATE\_FORMAT

log date format as used by the logging module.

--log-file=LOG\_FILE path to a file when logging will be written to.

--log-file-level=LOG\_FILE\_LEVEL

log file logging level.

--log-file-format=LOG\_FILE\_FORMAT

log format as used by the logging module.

--log-file-date-format=LOG\_FILE\_DATE\_FORMAT

log date format as used by the logging module.

coverage reporting with distributed testing support:

--cov=[path] measure coverage for filesystem path (multi-allowed)

--cov-report=type type of report to generate: term, term-missing,

annotate, html, xml (multi-allowed). term, term-

missing may be followed by ":skip-covered". annotate,

html and xml may be followed by ":DEST" where DEST

specifies the output location. Use --cov-report= to

not generate any output.

--cov-config=path config file for coverage, default: .coveragerc

--no-cov-on-fail do not report coverage if test run fails, default:

False

--no-cov Disable coverage report completely (useful for

debuggers) default: False

--cov-fail-under=MIN Fail if the total coverage is less than MIN.

--cov-append do not delete coverage but append to current, default:

False

--cov-branch Enable branch coverage.

custom options:

--remote-data=[REMOTE\_DATA]

run tests with online data

--open-files fail if any test leaves files open

--doctest-plus enable running doctests with additional features not

found in the normal doctest plugin

--doctest-rst enable running doctests in .rst documentation

[pytest] ini-options in the first pytest.ini|tox.ini|setup.cfg file found:

markers (linelist) markers for test functions

empty\_parameter\_set\_mark (string) default marker for empty parametersets

norecursedirs (args) directory patterns to avoid for recursion

testpaths (args) directories to search for tests when no files or directories are given in the command line.

console\_output\_style (string) console output: classic or with additional progress information (classic|progress).

usefixtures (args) list of default fixtures to be used with this project

python\_files (args) glob-style file patterns for Python test module discovery

python\_classes (args) prefixes or glob names for Python test class discovery

python\_functions (args) prefixes or glob names for Python test function and method discovery

xfail\_strict (bool) default for the strict parameter of xfail markers when not given explicitly (default: False)

junit\_suite\_name (string) Test suite name for JUnit report

junit\_logging (string) Write captured log messages to JUnit report: one of no|system-out|system-err

doctest\_optionflags (args) option flags for doctests

doctest\_encoding (string) encoding used for doctest files

cache\_dir (string) cache directory path.

filterwarnings (linelist) Each line specifies a pattern for warnings.filterwarnings. Processed after -W and --pythonwarnings.

log\_print (bool) default value for --no-print-logs

log\_level (string) default value for --log-level

log\_format (string) default value for --log-format

log\_date\_format (string) default value for --log-date-format

log\_cli (bool) enable log display during test run (also known as "live logging").

log\_cli\_level (string) default value for --log-cli-level

log\_cli\_format (string) default value for --log-cli-format

log\_cli\_date\_format (string) default value for --log-cli-date-format

log\_file (string) default value for --log-file

log\_file\_level (string) default value for --log-file-level

log\_file\_format (string) default value for --log-file-format

log\_file\_date\_format (string) default value for --log-file-date-format

addopts (args) extra command line options

minversion (string) minimally required pytest version

remote\_data\_strict (bool) If 'True', tests will fail if they attempt to access the internet but are not explicitly marked with 'remote\_data'

open\_files\_ignore (args) when used with the --open-files option, allows specifying names of files that may be ignored when left open between tests--files in this list are matched may be

doctest\_plus (string) enable running doctests with additional features not found in the normal doctest plugin

doctest\_norecursedirs (args) like the norecursedirs option but applies only to doctest collection

doctest\_rst (string) Run the doctests in the rst documentation

environment variables:

PYTEST\_ADDOPTS extra command line options

PYTEST\_PLUGINS comma-separated plugins to load during startup

PYTEST\_DISABLE\_PLUGIN\_AUTOLOAD set to disable plugin auto-loading

PYTEST\_DEBUG set to enable debug tracing of pytest's internals

to see available markers type: pytest --markers

to see available fixtures type: pytest --fixtures

(shown according to specified file\_or\_dir or current dir if not specified; fixtures with leading '\_' are only shown with the '-v' option